

ANALISIS DE PRECIO DE CASAS

Importando librerias

```
import pandas as pd
datos=pd.read_csv("train.csv")
datos.head(20)
```

	Id	MSSubClass	MSZoning	LotFrontage	LotArea	Street	Alley
LotShape \							
0	1	60	RL	65.0	8450	Pave	NaN
Reg							
1	2	20	RL	80.0	9600	Pave	NaN
Reg							
2	3	60	RL	68.0	11250	Pave	NaN
IR1							
3	4	70	RL	60.0	9550	Pave	NaN
IR1							
4	5	60	RL	84.0	14260	Pave	NaN
IR1							
5	6	50	RL	85.0	14115	Pave	NaN
IR1							
6	7	20	RL	75.0	10084	Pave	NaN
Reg							
7	8	60	RL	NaN	10382	Pave	NaN
IR1							
8	9	50	RM	51.0	6120	Pave	NaN
Reg							
9	10	190	RL	50.0	7420	Pave	NaN
Reg							
10	11	20	RL	70.0	11200	Pave	NaN
Reg							
11	12	60	RL	85.0	11924	Pave	NaN
IR1							
12	13	20	RL	NaN	12968	Pave	NaN
IR2							
13	14	20	RL	91.0	10652	Pave	NaN
IR1							
14	15	20	RL	NaN	10920	Pave	NaN
IR1							
15	16	45	RM	51.0	6120	Pave	NaN
Reg							
16	17	20	RL	NaN	11241	Pave	NaN
IR1							
17	18	90	RL	72.0	10791	Pave	NaN
Reg							
18	19	20	RL	66.0	13695	Pave	NaN
Reg							
19	20	20	RL	70.0	7560	Pave	NaN

Reg							
LandContour		Utilities	...	PoolArea	PoolQC	Fence	MiscFeature
MiscVal	\						
0	Lvl	AllPub	...	0	NaN	NaN	NaN
0							
1	Lvl	AllPub	...	0	NaN	NaN	NaN
0							
2	Lvl	AllPub	...	0	NaN	NaN	NaN
0							
3	Lvl	AllPub	...	0	NaN	NaN	NaN
0							
4	Lvl	AllPub	...	0	NaN	NaN	NaN
0							
5	Lvl	AllPub	...	0	NaN	MnPrv	Shed
700							
6	Lvl	AllPub	...	0	NaN	NaN	NaN
0							
7	Lvl	AllPub	...	0	NaN	NaN	Shed
350							
8	Lvl	AllPub	...	0	NaN	NaN	NaN
0							
9	Lvl	AllPub	...	0	NaN	NaN	NaN
0							
10	Lvl	AllPub	...	0	NaN	NaN	NaN
0							
11	Lvl	AllPub	...	0	NaN	NaN	NaN
0							
12	Lvl	AllPub	...	0	NaN	NaN	NaN
0							
13	Lvl	AllPub	...	0	NaN	NaN	NaN
0							
14	Lvl	AllPub	...	0	NaN	GdWo	NaN
0							
15	Lvl	AllPub	...	0	NaN	GdPrv	NaN
0							
16	Lvl	AllPub	...	0	NaN	NaN	Shed
700							
17	Lvl	AllPub	...	0	NaN	NaN	Shed
500							
18	Lvl	AllPub	...	0	NaN	NaN	NaN
0							
19	Lvl	AllPub	...	0	NaN	MnPrv	NaN
0							
MoSold	YrSold	SaleType	SaleCondition	SalePrice			
0	2	2008	WD	Normal			
1	5	2007	WD	Normal			
2	9	2008	WD	Normal			

3	2	2006	WD	Abnorml	140000
4	12	2008	WD	Normal	250000
5	10	2009	WD	Normal	143000
6	8	2007	WD	Normal	307000
7	11	2009	WD	Normal	200000
8	4	2008	WD	Abnorml	129900
9	1	2008	WD	Normal	118000
10	2	2008	WD	Normal	129500
11	7	2006	New	Partial	345000
12	9	2008	WD	Normal	144000
13	8	2007	New	Partial	279500
14	5	2008	WD	Normal	157000
15	7	2007	WD	Normal	132000
16	3	2010	WD	Normal	149000
17	10	2006	WD	Normal	90000
18	6	2008	WD	Normal	159000
19	5	2009	COD	Abnorml	139000

[20 rows x 81 columns]

Tamaño del Data Frame

datos.shape

(1460, 81)

datos["Id"]

0	1
1	2
2	3
3	4
4	5
	...
1455	1456
1456	1457
1457	1458
1458	1459
1459	1460

Name: Id, Length: 1460, dtype: int64

ID de las casas frente a su PRECIO

datos[["Id", "SalePrice"]]

	Id	SalePrice
0	1	208500
1	2	181500
2	3	223500
3	4	140000
4	5	250000

```

...      ...      ...
1455  1456      175000
1456  1457      210000
1457  1458      266500
1458  1459      142125
1459  1460      147500

```

```
[1460 rows x 2 columns]
```

El precio promedio de los precio de venta

```
datos['SalePrice'].mean()
```

```
np.float64(180921.19589041095)
```

Descripcion total de todos los datos del data frame

```
datos.describe()
```

```

              Id  MSSubClass  LotFrontage  LotArea
OverallQual \
count  1460.000000  1460.000000  1201.000000  1460.000000
1460.000000
mean    730.500000    56.897260    70.049958  10516.828082
6.099315
std     421.610009    42.300571    24.284752    9981.264932
1.382997
min       1.000000    20.000000    21.000000    1300.000000
1.000000
25%     365.750000    20.000000    59.000000    7553.500000
5.000000
50%     730.500000    50.000000    69.000000    9478.500000
6.000000
75%    1095.250000    70.000000    80.000000   11601.500000
7.000000
max    1460.000000   190.000000   313.000000  215245.000000
10.000000

```

```

      OverallCond  YearBuilt  YearRemodAdd  MasVnrArea
BsmtFinSF1 ... \
count  1460.000000  1460.000000  1460.000000  1452.000000
1460.000000 ...
mean     5.575342  1971.267808  1984.865753   103.685262
443.639726 ...
std      1.112799   30.202904   20.645407   181.066207
456.098091 ...
min       1.000000  1872.000000  1950.000000    0.000000
0.000000 ...
25%       5.000000  1954.000000  1967.000000    0.000000
0.000000 ...
50%       5.000000  1973.000000  1994.000000    0.000000

```

```

383.500000 ...
75%      6.000000  2000.000000  2004.000000  166.000000
712.250000 ...
max      9.000000  2010.000000  2010.000000  1600.000000
5644.000000 ...

      WoodDeckSF  OpenPorchSF  EnclosedPorch  3SsnPorch
ScreenPorch \
count  1460.000000  1460.000000  1460.000000  1460.000000
1460.000000
mean    94.244521   46.660274   21.954110   3.409589
15.060959
std    125.338794   66.256028   61.119149   29.317331
55.757415
min      0.000000    0.000000    0.000000    0.000000
0.000000
25%      0.000000    0.000000    0.000000    0.000000
0.000000
50%      0.000000   25.000000    0.000000    0.000000
0.000000
75%     168.000000   68.000000    0.000000    0.000000
0.000000
max     857.000000  547.000000  552.000000  508.000000
480.000000

      PoolArea      MiscVal      MoSold      YrSold
SalePrice
count  1460.000000  1460.000000  1460.000000  1460.000000
1460.000000
mean    2.758904   43.489041   6.321918  2007.815753
180921.195890
std    40.177307  496.123024   2.703626   1.328095
79442.502883
min      0.000000    0.000000    1.000000  2006.000000
34900.000000
25%      0.000000    0.000000    5.000000  2007.000000
129975.000000
50%      0.000000    0.000000    6.000000  2008.000000
163000.000000
75%      0.000000    0.000000    8.000000  2009.000000
214000.000000
max     738.000000  15500.000000  12.000000  2010.000000
755000.000000

[8 rows x 38 columns]

```

IMPORTANDO LIBRERIAS

```

import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
import numpy as np
from scipy import stats

from sklearn.preprocessing import StandardScaler

import warnings
%matplotlib inline

from scipy.stats import norm

```

CUALES COLUMNAS TIENEN MAYOR IMPACTO

PRECIO DE VENTA

Descripcion general del precio de venta

```
datos["SalePrice"].describe()
```

```

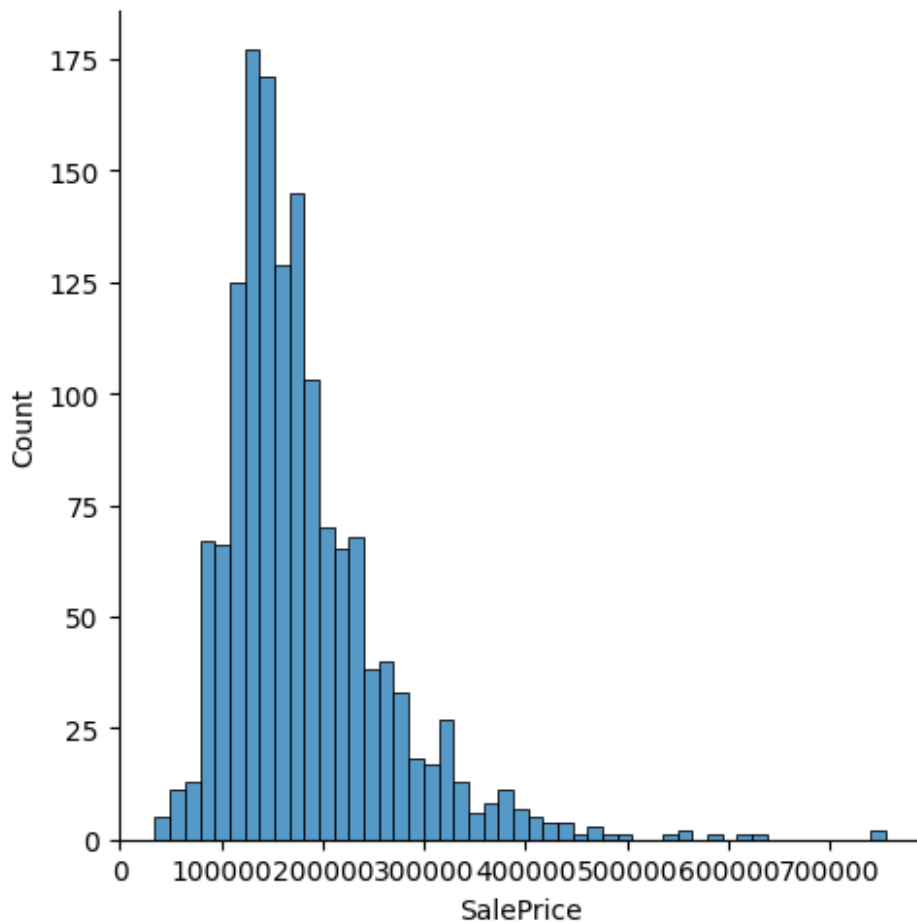
count      1460.000000
mean       180921.195890
std        79442.502883
min        34900.000000
25%       129975.000000
50%       163000.000000
75%       214000.000000
max        755000.000000
Name: SalePrice, dtype: float64

```

Grafica de Precio de venta frente a cantidad de las casas

```
sns.displot(datos["SalePrice"])
```

```
<seaborn.axisgrid.FacetGrid at 0x21124478c10>
```



cuando no esta bien distribuido se llama curtosis

```
datos["SalePrice"].skew()
np.float64(1.8828757597682129)
datos["SalePrice"].kurt()
np.float64(6.536281860064529)
```

AREA DE VIVIENDA

var="GrLivArea"

Generar tabla de precio de venta frente al area de vivienda

```
data=pd.concat([datos["SalePrice"],datos[var]], axis=1)
data.head(20)
```

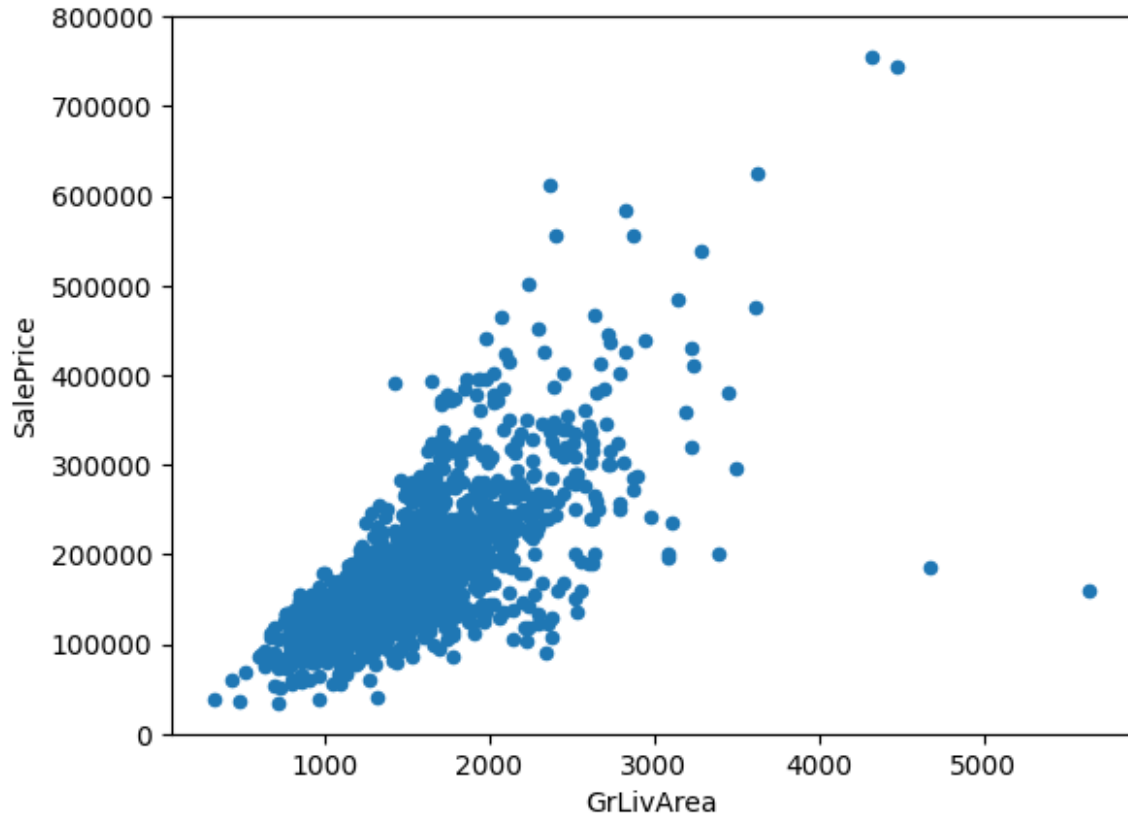
	SalePrice	GrLivArea
0	208500	1710
1	181500	1262

2	223500	1786
3	140000	1717
4	250000	2198
5	143000	1362
6	307000	1694
7	200000	2090
8	129900	1774
9	118000	1077
10	129500	1040
11	345000	2324
12	144000	912
13	279500	1494
14	157000	1253
15	132000	854
16	149000	1004
17	90000	1296
18	159000	1114
19	139000	1339

Grafica de Precio de venta frente a area de vivienda

```
data.plot.scatter(x=var,y="SalePrice",ylim=(0,800000))
```

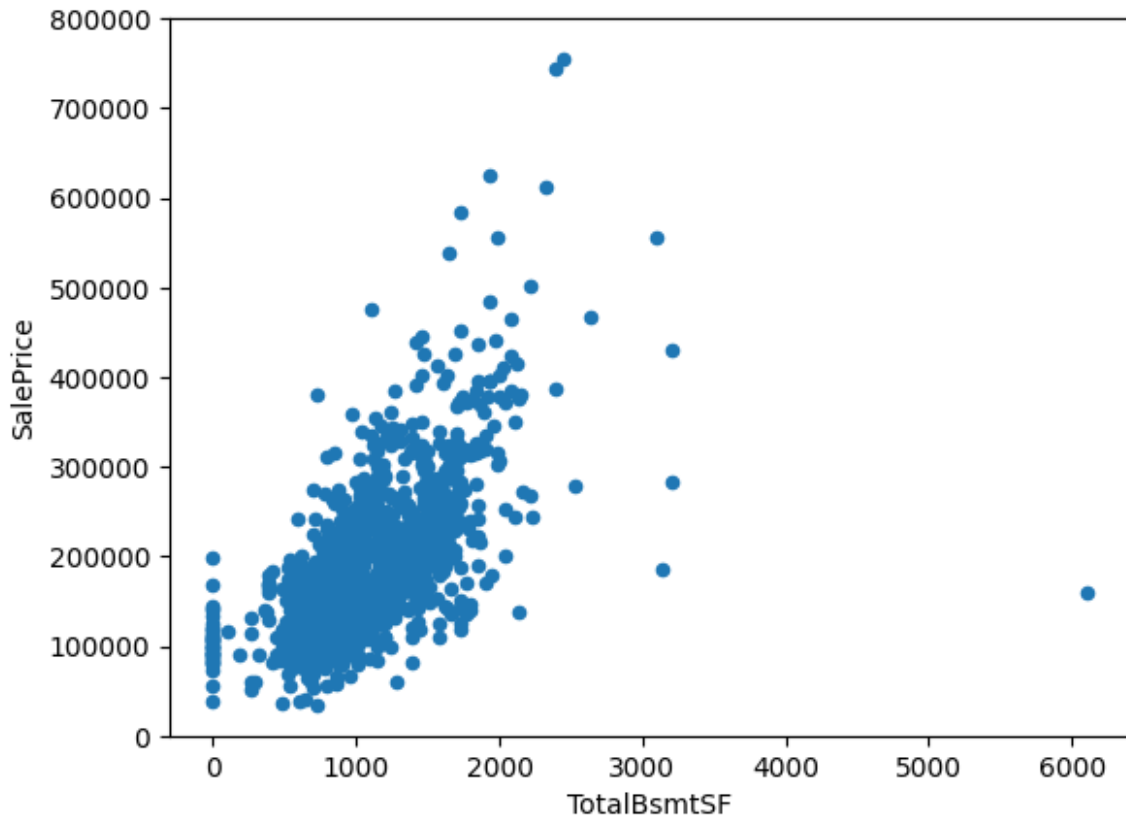
```
<Axes: xlabel='GrLivArea', ylabel='SalePrice'>
```



AREA TOTAL DEL SOTANO

Grafica del area del sotano frente al precio de vivienda.

```
var="TotalBsmtSF"
data=pd.concat([datos["SalePrice"],datos[var]], axis=1)
data.plot.scatter(x=var,y="SalePrice",ylim=(0,800000))
<Axes: xlabel='TotalBsmtSF', ylabel='SalePrice'>
```



RELACIONES CATEGORICAS

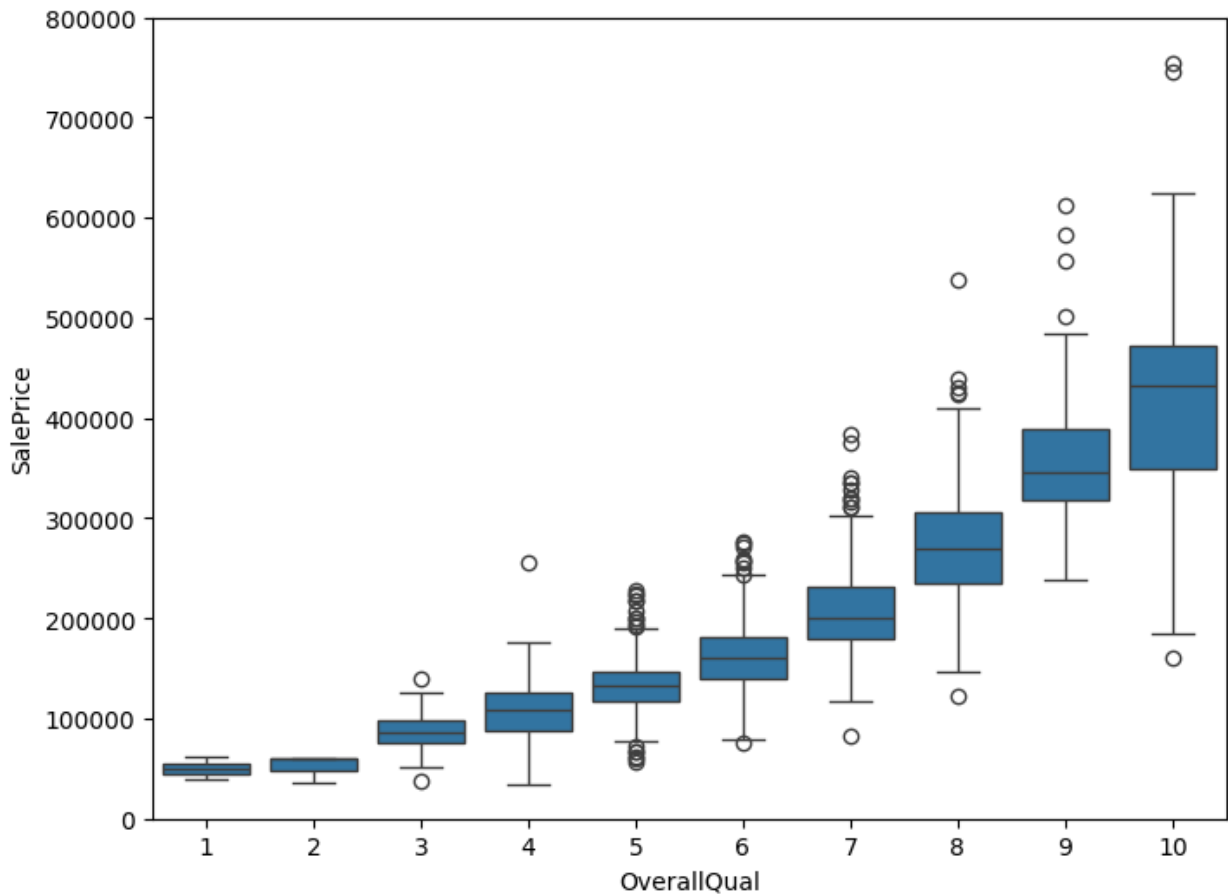
CALIDAD DE CASA

Grafica de caja y bigotes de la calidad de la casa frente al precio de venta

```
var="OverallQual"
data=pd.concat([datos["SalePrice"],datos[var]], axis=1)

f, ax=plt.subplots(figsize=(8,6))
fig=sns.boxplot(x=var,y="SalePrice",data=data)
fig.axis(ymin=0,ymax=800000)
```

```
(np.float64(-0.5), np.float64(9.5), np.float64(0.0),
np.float64(800000.0))
```



AÑO DE CONSTRUCCION

Grafica del precio de venta frente al año de construccion

```
var="YearBuilt"
data=pd.concat([datos["SalePrice"],datos[var]], axis=1)
f, ax=plt.subplots(figsize=(20,8))
fig=sns.boxplot(x=var,y="SalePrice",data=data)
fig.axis(ymin=0,ymax=800000)
plt.xticks(rotation=90)
```

```
([0,
1,
2,
3,
4,
5,
6,
7,
```

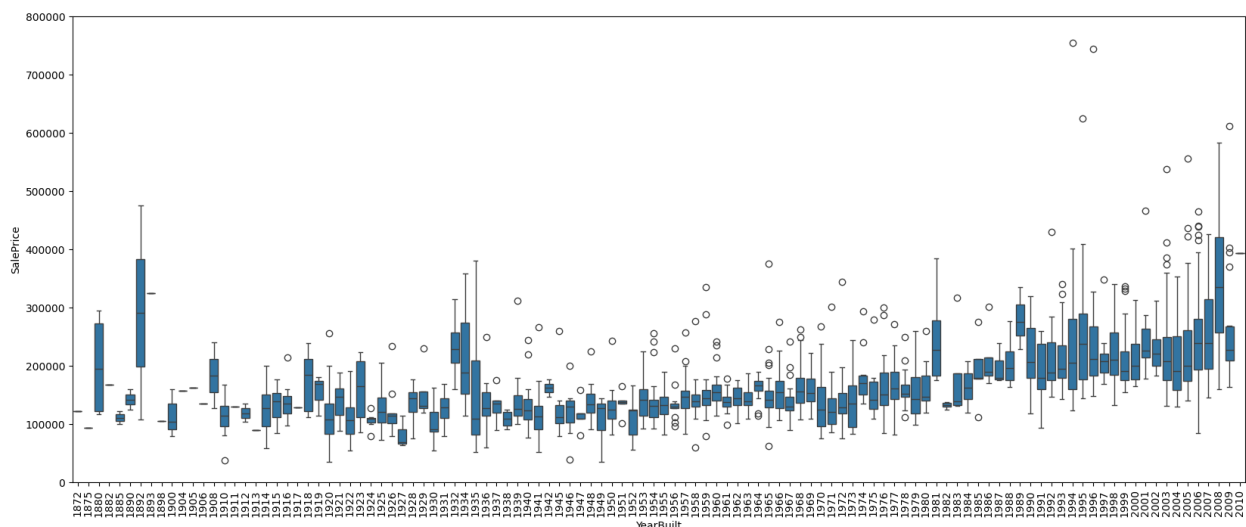
8,
9,
10,
11,
12,
13,
14,
15,
16,
17,
18,
19,
20,
21,
22,
23,
24,
25,
26,
27,
28,
29,
30,
31,
32,
33,
34,
35,
36,
37,
38,
39,
40,
41,
42,
43,
44,
45,
46,
47,
48,
49,
50,
51,
52,
53,
54,
55,
56,

57,
58,
59,
60,
61,
62,
63,
64,
65,
66,
67,
68,
69,
70,
71,
72,
73,
74,
75,
76,
77,
78,
79,
80,
81,
82,
83,
84,
85,
86,
87,
88,
89,
90,
91,
92,
93,
94,
95,
96,
97,
98,
99,
100,
101,
102,
103,
104,
105,

```
106,  
107,  
108,  
109,  
110,  
111],  
[Text(0, 0, '1872'),  
Text(1, 0, '1875'),  
Text(2, 0, '1880'),  
Text(3, 0, '1882'),  
Text(4, 0, '1885'),  
Text(5, 0, '1890'),  
Text(6, 0, '1892'),  
Text(7, 0, '1893'),  
Text(8, 0, '1898'),  
Text(9, 0, '1900'),  
Text(10, 0, '1904'),  
Text(11, 0, '1905'),  
Text(12, 0, '1906'),  
Text(13, 0, '1908'),  
Text(14, 0, '1910'),  
Text(15, 0, '1911'),  
Text(16, 0, '1912'),  
Text(17, 0, '1913'),  
Text(18, 0, '1914'),  
Text(19, 0, '1915'),  
Text(20, 0, '1916'),  
Text(21, 0, '1917'),  
Text(22, 0, '1918'),  
Text(23, 0, '1919'),  
Text(24, 0, '1920'),  
Text(25, 0, '1921'),  
Text(26, 0, '1922'),  
Text(27, 0, '1923'),  
Text(28, 0, '1924'),  
Text(29, 0, '1925'),  
Text(30, 0, '1926'),  
Text(31, 0, '1927'),  
Text(32, 0, '1928'),  
Text(33, 0, '1929'),  
Text(34, 0, '1930'),  
Text(35, 0, '1931'),  
Text(36, 0, '1932'),  
Text(37, 0, '1934'),  
Text(38, 0, '1935'),  
Text(39, 0, '1936'),  
Text(40, 0, '1937'),  
Text(41, 0, '1938'),  
Text(42, 0, '1939'),
```

```
Text(43, 0, '1940'),
Text(44, 0, '1941'),
Text(45, 0, '1942'),
Text(46, 0, '1945'),
Text(47, 0, '1946'),
Text(48, 0, '1947'),
Text(49, 0, '1948'),
Text(50, 0, '1949'),
Text(51, 0, '1950'),
Text(52, 0, '1951'),
Text(53, 0, '1952'),
Text(54, 0, '1953'),
Text(55, 0, '1954'),
Text(56, 0, '1955'),
Text(57, 0, '1956'),
Text(58, 0, '1957'),
Text(59, 0, '1958'),
Text(60, 0, '1959'),
Text(61, 0, '1960'),
Text(62, 0, '1961'),
Text(63, 0, '1962'),
Text(64, 0, '1963'),
Text(65, 0, '1964'),
Text(66, 0, '1965'),
Text(67, 0, '1966'),
Text(68, 0, '1967'),
Text(69, 0, '1968'),
Text(70, 0, '1969'),
Text(71, 0, '1970'),
Text(72, 0, '1971'),
Text(73, 0, '1972'),
Text(74, 0, '1973'),
Text(75, 0, '1974'),
Text(76, 0, '1975'),
Text(77, 0, '1976'),
Text(78, 0, '1977'),
Text(79, 0, '1978'),
Text(80, 0, '1979'),
Text(81, 0, '1980'),
Text(82, 0, '1981'),
Text(83, 0, '1982'),
Text(84, 0, '1983'),
Text(85, 0, '1984'),
Text(86, 0, '1985'),
Text(87, 0, '1986'),
Text(88, 0, '1987'),
Text(89, 0, '1988'),
Text(90, 0, '1989'),
Text(91, 0, '1990'),
```

```
Text(92, 0, '1991'),
Text(93, 0, '1992'),
Text(94, 0, '1993'),
Text(95, 0, '1994'),
Text(96, 0, '1995'),
Text(97, 0, '1996'),
Text(98, 0, '1997'),
Text(99, 0, '1998'),
Text(100, 0, '1999'),
Text(101, 0, '2000'),
Text(102, 0, '2001'),
Text(103, 0, '2002'),
Text(104, 0, '2003'),
Text(105, 0, '2004'),
Text(106, 0, '2005'),
Text(107, 0, '2006'),
Text(108, 0, '2007'),
Text(109, 0, '2008'),
Text(110, 0, '2009'),
Text(111, 0, '2010')])
```



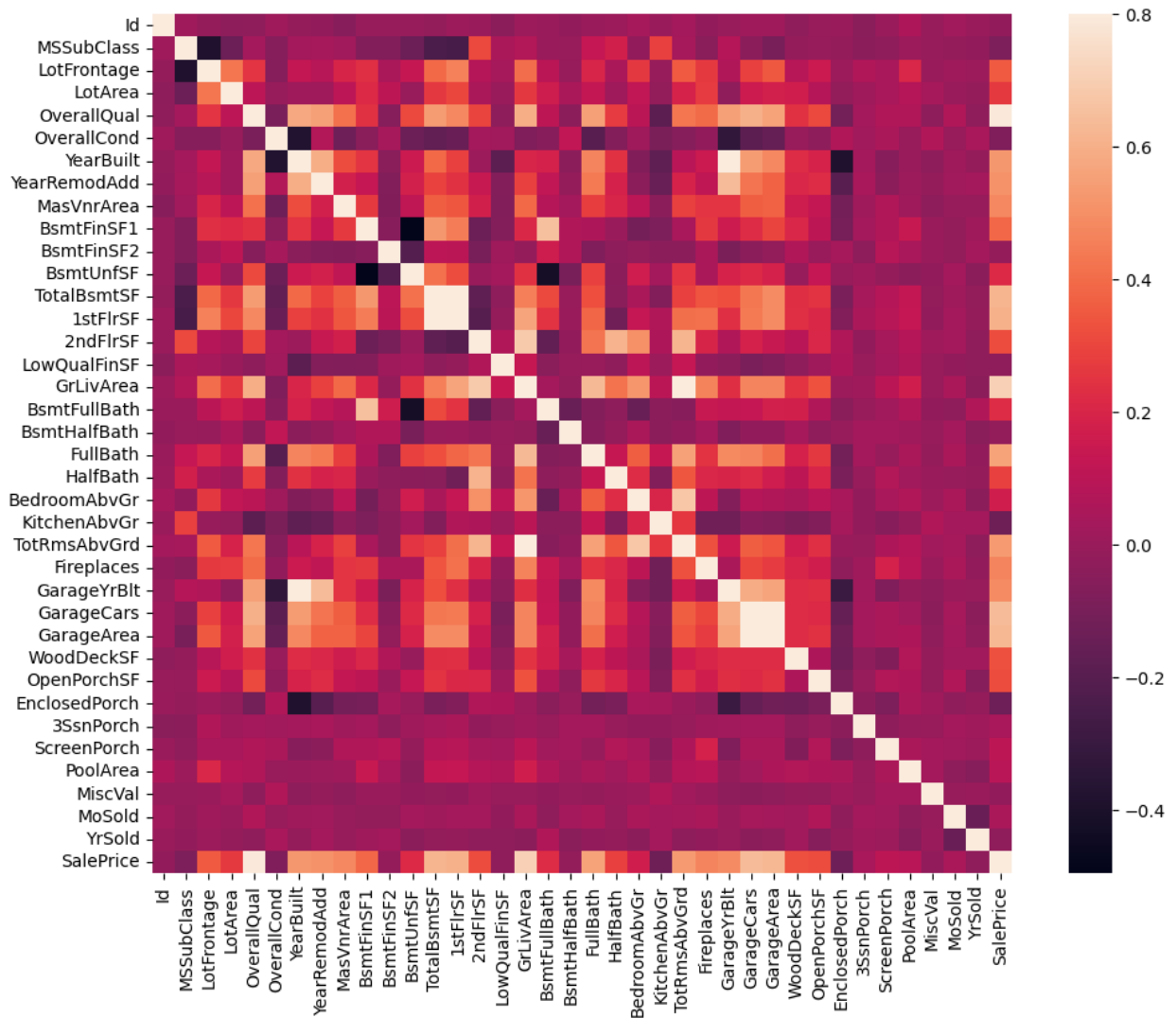
MATRIZ DE CORRELACION

Grafica de correlacion donde se puede visualizar cuales variables influyen mas frente al precio de venta

```
corrmat = datos.corr(numeric_only = [False])
```

```
f,ax=plt.subplots(figsize=(12,9))
sns.heatmap(corrmat,vmax=.8,square=True)
```

<Axes: >



Observamos que las variables de Area de vivienda, calidad, area del garage, tamaño del garage son las variables que mas modifican influyen en el precio de venta.